An Energy Efficient Clique Based Clustering and Routing Mechanism in Wireless Sensor Networks

Kamanashis Biswas, Vallipuram Muthukkumarasamy, Elankayer Sithirasanen, Muhammad Usman

*School of Information and Communication Technology, Griffith University
Gold Coast, Australia*

kamanashis.biswas@griffithuni.edu.au

v.muthu@griffith.edu.au

e.sithirasanen@griffith.edu.au

muhammad.usman3@griffithuni.edu.au

Abstract—In this paper we propose an energy efficient clique based routing scheme for wireless sensor networks. One of the major issues in wireless sensor networks is developing an energy efficient routing protocol. Since the sensor nodes are small-sized battery operated devices, energy conservation is considered a critical issue to maximize network lifetime. Our proposed routing protocol partitions sensor networks into several disjoint cliques where each node is in single hop distance from all of its neighbour nodes. At the beginning of the clustering process, each node obtains a list of its neighbours' connectivity as well as their degree of connection. Then, the node with highest degree of connection initiates clique formation process and makes the cluster. Among all the members of a cluster, the node with maximum energy is selected as cluster head. Instead of rotating cluster heads in every round, CH rotation depends on threshold value which is obtained by averaging residual energy of all the nodes in a cluster and then making it half. Simulation results with MATLAB show that our proposed protocol is effective in terms of network lifetime as well as total energy dissipation.

I. INTRODUCTION

Continued advances in micro-electronic-mechanical-systems (MEMS), smart sensors and wireless communication technologies have enabled the deployment of large scale wireless sensor networks. The potential applications of wireless sensor networks are highly varied, such as disaster management situations (e.g. earthquakes [1]), object tracking in battle field, habitant monitoring [2] etc. Sensors in such a network are often equipped with limited computational capability, memory and transmission range. Due to the sensors' limited power, the network lifetime is a major concern; specially, for applications of wireless sensor networks in harsh environments. Hence, energy efficient routing mechanisms and adaptive clustering schemes are developed to prolong the network lifetime.

In this paper, we propose a CBCR (Clique Based Clustering and Routing) protocol to minimize the energy dissipation in sensor networks. It is a clustering based protocol that forms non-overlapping clusters of size n , where n is the maximum cluster size. By exchanging information of 1-hop neighbors, all sensor nodes in the network are grouped into a number of disjoint cliques, in which all the nodes can directly

communicate with each other. Among all the nodes in a cluster, the node with maximum energy becomes cluster head. The key features of CBCR are: self-configuration and localized coordination, maximum energy cluster head, periodical rotation of cluster head, hierarchical forwarding, load balance, fault tolerance and scalability.

II. RELATED WORK

Low Energy Adaptive Clustering Hierarchy (LEACH) is a clustering-based protocol [3]. LEACH forms clusters on the basis of the received signal strength and uses the CH nodes as routers to communicate with base stations. It uses a function to determine the probability p of a node that wants to be a cluster head (CH). A candidate node with probability p broadcasts its decision of candidature to other nodes. Each non-CH node finds out its cluster by selecting the CH that can be reached by using the lowest communication energy. LEACH also performs periodical rotation of cluster heads in order to balance the load. The main problem of LEACH is that the selection of cluster head is probabilistic. Hence, there is a good chance that a node with very low energy gets selected as a CH. When this node dies, the whole cluster becomes dysfunctional.

Threshold sensitive Energy Efficient Sensor Network protocol (TEEN) and Power-Efficient Gathering in Sensor Information Systems (PEGASIS) are two energy efficient protocols proposed to improve the performance of LEACH. TEEN is the first protocol developed for reactive networks and used in temperature sensing applications [4]. It has two additional restrictions. First, when the absolute value of the sensed attribute is beyond a Hard Threshold (H_T), the node sensing this value must switch on its transmitter and report it. Second, when a change in the value of the sensed attribute is larger than a Soft Threshold (S_T), it triggers the node to switch on its transmitter and report the sensed data. A node will report data only when the sensed value exceeds H_T or the value's change is larger than S_T . The main drawbacks of this scheme are that it is not well suited for applications where the user needs to get data on a regular basis and practical implementation would have to ensure that there are no collisions in the cluster. TDMA scheduling of the nodes can be

used to avoid this problem but this causes a delay in the reporting of the time-critical data.

PAGASIS is a chain based protocol which reduces energy dissipation and elongates network lifetime by making nodes communicate only with nearest neighbors [5]. It assumes that each node must know location information about all other nodes at first. PAGASIS starts with the farthest node from the base station (BS). The chain can be constructed easily by using greedy algorithm. The chain leader aggregates data and forwards it to the BS. Each node in the chain takes turn to be the leader in order to balance the overhead involved in communication between the BS and the chain leader. The drawback of PAGASIS is that the protocol makes certain assumptions that are hard to realize. Firstly, each node is capable of communicating with BS, whereas sensor nodes generally use multi-hop routing. Secondly, all nodes have equal energy and die at the same time. Finally, excessive data delay is another limitation of PAGASIS.

Power Efficient and Adaptive Clustering Hierarchy (PEACH) protocol forms cluster without any additional overhead, by using overhearing characteristics of wireless communication [6]. By overhearing a node gets information of the source and destination of packets send by its neighbor nodes. Overheard information is then used to form cluster and hence avoids exchange of control messages and packet transmission overhead such as advertisement, announcement, joining, and scheduling messages. In order to provide an adaptive multi-level clustering, it uses probabilistic routing protocols. PEACH assumes that all sensor nodes have equal capabilities. This assumption is hard for WSN applications.

Like TEEN and PAGASIS, Maximum Energy Cluster Head (MECH) routing protocol is proposed to improve the performance of LEACH [7]. MECH constructs clusters based on radio range and the number of cluster members. In this scheme, each node broadcasts a hello message to its neighbors and each node records the number of neighbors. When the number of neighbors reach to a predefined cluster number, then the node claim itself as CH and broadcasts an advertisement to its one hop neighbors. The forwarding phase of MECH is guided by the BS to ensure multi-hop routing. MECH ensures that there will not be more than one cluster head in a certain transmission range. The main problem of this protocol is that initial CH selection process may select a node with low energy as CH.

Besides these routing schemes, a number of energy efficient protocols are proposed and developed by the researchers over the year. It is noticeable that in most of the energy efficient routing algorithms [3, 8, 9], CH selection process is probabilistic rather than deterministic. Similarly, few clique based clustering schemes are also proposed (e.g. [10, 11]), which mainly emphasize on routing and security issues. The authors in [12], propose an energy-efficient clique based geocast algorithm and the cluster head in this scheme is located in the central area of the cluster to minimize energy consumption. This is really hard to achieve and if the node in central location possess low energy then the node will die fast.

In this paper, we present a distributed routing protocol which is different from the above mentioned clique based protocols. CBCR forms clusters by partitioning the networks into a number of disjoint cliques and selects the maximum energy node as CH. In [10-12], every node computes its local maximum clique and then interchange the information to form global maximum clique. This approach consumes more time, memory and communication resources. But, in CBCR only the nodes with highest degree of connection can start clique formation process and if a node is included in a cluster by its neighbor, it will never get the chance to initiate clustering process. Moreover, instead of CH rotation at each round, the energy level of the current CH is compared with a threshold value to prolong the network lifetime in CBCR. Thus, CBCR ensures low energy dissipation and long system lifetime for wireless sensor networks.

III. ENERGY EFFICIENT CLIQUE BASED CLUSTERING AND ROUTING PROTOCOL(CBCR)

The network model used in this paper and some considerations of developing an efficient clustering protocol are described. Then, the CBCR protocol is presented with detailed examples.

A. Network Model

This section considers the wireless sensor networks consisting of hundreds or thousands of deployed sensor nodes in the sensing field. It is assumed by the following properties of the WSNs to simplify the network model.

- Every sensor node is assigned a unique node ID.
- Links are bidirectional, i.e. two nodes u_1 and u_2 are connected if both of them can communicate to each other using the same transmission power.
- All sensor nodes have limited batteries, and to recharge the batteries is impossible. Efficient energy-aware clustering protocols require for energy conservation.
- The network consists of multiple mobile/stationary nodes, which implies that energy consumption is not uniform for all nodes.
- All sensor nodes have various transmission power levels, and each node can change the power level dynamically.
- Network topology does not change during cluster formation process.

We use the similar model described in [3] for the radio hardware energy dissipation where the transmitter dissipates energy to run the radio electronics and the power amplifier, and the receiver dissipates energy to run the radio electronics only. Both the free space (d^2 power loss) and multipath fading (d^4 power loss) channel models are used depending on the distance between the transmitter and receiver. Power control can be used to invert this loss by appropriately setting the power amplifier if the distance is less than a threshold (d_0), the free space (f_s) model is used; otherwise, the multipath (mp) model is used. Thus, to transmit a k -bit message over a distance d , the radio expends:

$$E_{Tx}(k,d) = \begin{cases} k \times E_{elec} + k \times \epsilon_{fs} d^2, & d < d_0 \\ k \times E_{elec} + k \times \epsilon_{mp} d^4, & d \geq d_0 \end{cases} \quad (1)$$

and to receive this message, the radio expends:

$$E_{Rx}(k) = k \times E_{elec} \quad (2)$$

The electronics energy, E_{elec} , depends on various factors such as the digital coding, modulation, filtering, and spreading of the signal whereas the amplifier energy $\epsilon_{fs} d^2$ or $\epsilon_{mp} d^4$ depends on the distance to the receiver and the acceptable bit-error rate.

B. The CBCR Protocol

In this section, we describe the proposed CBCR protocol. Our protocol is divided into four phases: cluster formation phase, set-up phase, steady phase and forwarding phase. Following is the detailed explanation of the proposed CBCR protocol.

1) *Cluster Formation Phase*: Each node receives the connectivity list from its all one-hop neighbours and computes the degree of connection from that list. In addition to this information, the received signal strength of each neighbour is also measured and saved initially. The *algorithm 1* describes the connectivity matrix formation process. Here, we assume that node m is forming the connectivity matrix. Node m receives neighbour node lists from each of its neighbours and finds out the common connection of each of them with itself.

Algorithm 1: Formation of Connectivity Matrix

INPUT: NNL_i // NNL denotes neighbour node list of node i
 // Here, i is the neighbour of node m
OUTPUT: G_m // Connectivity matrix of node m
STEPS:
 $DoC_i = \text{NoE}(NNL_i)$; // Degree of connectivity equals
 // to number of elements in NNL
for each neighbour node (i) of node m
for $k = 1$ to DoC_m
if $NNL_{i,k} == NLL_{m,k}$
 Add $NNL_{i,k}$ to G_m
end_if
end_for
 Add DoC_i and RSS_i to G_m
end_for

In this way, each node forms its connectivity matrix which includes common connections of neighbour nodes, their degree of connectivity and energy level. On achieving such matrix, each node checks whether it holds highest degree of connection or not. If it does not, then it waits until it is added to a cluster by its neighbour nodes or it becomes the highest degree node after some iteration. Otherwise, it waits for random time t and starts cluster formation process.

It is well known that finding out maximum clique in a random graph is an NP-complete problem [13]. To reduce the

time complexity, we use greedy algorithm to compute maximal clique as shown in the *algorithm 2* [14]. Again, we assume that the algorithm is executed by node m as node m holds the highest degree of connections among all its neighbors. As soon as node m forms the cluster, it selects the cluster head and announces the message to the neighbor nodes. The message includes node ID of all the cluster members followed by CH ID.

Algorithm 2: Cluster Formation Process

INPUT: $G_m = \{V_i, E_i\}, i \in V_i$ // Connectivity matrix of node m
OUTPUT: C_m // Clique node list, initially holds m
STEPS:
 $CS = n$; // Set cluster size, it equals to node size by default
 $CN = m$; // Set current node to m
if G_m of node m is null,
 then return // indicates single node cluster
else
while (size of $C_m \leq CS$ && exists nodes in common)
 Find k , having maximum common connection with CN
 Add k to C_m
 $CN = k$;
end_while
end_if_else

Algorithm 3 describes the cluster head selection and announcement process by node m . Receiving this message, the nodes which have joined in a cluster, will inform its neighbors regarding its membership. Hence, the non-clustered nodes will be able to update their connectivity matrix list by removing the clustered nodes.

Algorithm 3: CH Selection and Announcement Process

INPUT: C_m // Connectivity matrix of node m
OUTPUT: $C_m\text{-info}$ // Cluster Head and cluster members information
STEPS:
if round == 1
 Select the node with maximum energy
 Announce $C_m\text{-info}$ to the cluster members
else
 $T_level = \text{avg_eng} / 2$; // T_level indicates threshold value
if $RSS(m) < T_level$
 $\text{max_eng} = RSS(m)$;
 $\text{flag} = 0$;
for each member (i) of C_m
if $\text{eng}(i) > \text{max_eng}$
 $\text{New_CH} = i$;
 $\text{Start_node} = C_m(1)$;
 $\text{flag} = 1$;
end_if
end_for
end_if
if $\text{flag} == 1$
 interchange between New_CH and Start_node
 announce $C_m\text{-info}$ to the cluster members
end_if
end_if

In the following, we explain cluster formation process in details. To facilitate the discussion, we use simple graph as shown in Fig. 1. It shows a sensor network consisting of six

sensor nodes and nine edges. A bidirectional edge from node i to node j indicates that both nodes are directly connected to each other.

Fig. 1: Sensor network with six nodes

In this network, when *node 1* receives the connectivity list $\{1, 2, 3\}$ from *node 5*, it ignores the link between *node 5* and *node 2* as *node 1* is not directly connected to *node 2*. But it does not decrease the degree of connectivity. Thus, each node forms its connectivity matrix using only common links.

TABLE I
CONNECTIVITY MATRIX OF NODE 1

CM of node 1	3	4	5	6	DoC
1	1	1	1	1	4
3	0	1	1	0	4
4	1	0	0	1	3
5	1	0	0	0	3
6	0	1	0	0	2

From connectivity matrix, each node knows whether it contains highest degree of connection or not. Any node containing equal or highest degree of connections can start cluster formation process after random time t to avoid collision. For example, in the above figure, both *node 1* and *node 3* have 4 connections. So, both of them are eligible for starting cluster formation process. But, one of them will be selected depending on the random timer. Suppose, *node 1* gets the chance, then *node 1* will inform all its neighbors about its clustering process. The neighbor nodes will wait for confirmation message from *node 1* to know whether it is selected or not to join in the cluster formed by *node 1*.

During this waiting period, if any other node attempts to initiate clustering process, the candidate node will receive a negative acknowledgement from the waiting node. As a result, the node trying to form new cluster will also go to wait state until it gets any feedback from that neighbor node. As soon as *node 1* has been formed its cluster, it will send a message ($C_{info}: \{1, 3, 4\}$) to its neighbors including the list of nodes who are nominated as members of that cluster. Each node selected to join in *node 1*'s cluster will inform all its neighbors so that they can remove the clustered nodes from their lists. It ensures that a node which is already joined in a cluster cannot be added to another cluster anymore. In this way, all the nodes will be member of a cluster. If a node does not have any node in its connectivity list, then the node itself form a cluster. For example, when *node 1* sends cluster information to *node 6*, *node 6* will discard *node 1* and *node 4* from its connectivity

list. As a result, it will be a single node without any link. Hence, *node 6* will form a single node cluster itself. The following figure shows the final output of the cluster formation process. The dotted circle in Fig. 2 denotes cluster heads selected on the basis of energy level.

Fig. 2: Cluster formation

2) *Set-up Phase*: Once CH is selected for a specific cluster and all the nodes in the cluster are informed about the CH, now, it is the responsibility of the selected CH to assign the time slots to all the members in the cluster. The CH can schedule the TDMA time slot for each cluster member at each round. In the set-up phase, every node will turn on the receiver as in LEACH. Then, the CH will broadcast an advertisement including the TDMA time slot information. Upon receiving the information, the cluster member will keep the transceiver off until its time slot. Thus, the cluster member will transmit the sensed data to the CH during its time slot. Besides, it also sends (piggy back) the residual energy to the CH. Hence, at the end of each round, CH has already received the residual energy of all the members in the cluster. After forwarding the data to the BS, the current CH selects the node with maximum residual energy as the new CH if the energy level of the current CH goes beyond the threshold limit (defined in algorithm 3). At the next round set-up phase, it will also broadcast the advertisement that contains the new CH in current round.

In TDMA scheme, time is divided into slots which are grouped into frames. We assume that there are n nodes in a cluster, as shown in Fig. 3, and each node is assigned a slot to transmit sensed data or control messages. In each frame one free slot is reserved for a new node joining the cluster. When a node decides to join in a cluster, the node uses this free slot to send its request. At the next round set-up phase, the CH will broadcast the joining request of the new node to all the nodes of the cluster. If no negative acknowledgement is received from any cluster member, then the node will be added by the CH and the TDMA frame will be readjusted.

Fig. 3: TDMA frame within a cluster

3) *Steady Phase*: Once the clusters are formed and the TDMA schedule is fixed, intra cluster communication can start. It involves sending sensed data and energy during the

allocated time slot to the cluster head. The cluster nodes adjust their transmission energy dynamically depending on the received signal strength of advertisement message. In this phase, if sensor node has sensed data it will wait for its allocated time slot to forward the data to the CH.

4) *Forwarding Phase*: Inter cluster communication or forwarding phase involves sending aggregated data from CH to the BS. As CBCR implements multihop routing, in this step, each CH sends data to other CH nearer to base station. We adopt the forwarding mechanism described in [7], in our proposed CBCR protocol.

In this scheme, the BS broadcasts a hello message periodically containing two parameters: *hop_count* (initially set to 0) and *energy_level* (set to ∞ initially). A non-cluster head node ignores this message. When receiving this message, a CH node adds one to the *hop_count* and replace the energy level with that of the CH, if it is higher than the energy level of current CH in the message. In next step, when another CH receives this message, it will record the previous CH 'x' and thus 'x' will be the forwarding CH to the base station for this CH. If a CH receives another hello message, it will make decisions according to the following rules:

Case 1: $hop_count_{old} < hop_count_{new}$, do nothing

Case 2: $hop_count_{old} > hop_count_{new}$, replace the forwarding CH

Case 3: $hop_count_{old} = hop_count_{new}$ and $energy_{old} < energy_{new}$, replace the forwarding CH, else do nothing.

IV. SIMULATION AND ANALYSIS

We simulated our proposed CBCR protocol described in Section-III using MATLAB and compared the results with LEACH which is the most popular and widely used energy efficient protocol. For our first experiment (i.e. energy dissipation in clustering process), we randomly deploy 100, 200, 300, 400 and 500 sensor nodes in a 100×100 (m²) simulation area. The other experiments are done with 100 nodes. The graphs presenting the simulation outcome is the average result of 1000 experiments. The simulation parameters used in the experiment are shown in Table 2.

TABLE III
PARAMETERS VALUES USED IN THE SIMULATION

Types	Parameters	Value
Network	Network Grid	From (0,0) to (100,100)
	Initial Energy	0.5 J
	Location of BS	(50, 50)
Application	Cluster Radius	20 m for CBCR
	Data Packet Size	128 bytes
	Broadcast Packet Size	24 bytes
	Packet Header Size	24 bytes
Radio Model	E_{elec}	50 nJ/bit
	E_{fs}	10 pJ/bit/m ²
	E_{mp}	0.0013 pJ/bit/m ⁴
	E_{DA}	5 nJ/bit/signal
	Threshold Distance (d_0)	87.71 m
Duration	Round Duration Time	15 s

Fig. 1(a): Initial energy dissipation

Fig. 1(b): Energy dissipation (10 R)

Fig. 1(c): Energy dissipation (20 R)

Fig. 1(d): Energy dissipation (30 R)

Fig. 1(a-d) shows the total energy dissipation in cluster formation process by LEACH and CBCR. Fig. 1(a) represents the initial energy consumption by these two protocols while Fig. 1(b), Fig 1(c) and Fig. 1(d) illustrates the amount of energy dissipated after 10, 20 and 30 rounds respectively. From the result, it is observed that total energy consumption in CBCR is higher than LEACH for about first 15 rounds. After initial round, it dissipates very low amount of energy than LEACH. This happens because CBCR requires transmitting more control messages to form clusters for the first time than LEACH. After that it only rotates the CH in each round on the basis of maximum energy. Hence, CBCR protocol has to send very few messages in each round. On the other hand, LEACH starts re-clustering at the beginning of each round. As a result, energy consumption in LEACH exceeds CBCR after few rounds.

The total number of nodes which are dead over simulation time of 2000 rounds is depicted in Fig. 2. It shows that nodes remain alive for longer time in CBCR than LEACH. Fig. 3 shows the total energy consumption of the network over simulation time. Based on simulation results, we find that LEACH dissipates more energy than CBCR due to re-clustering process.

Fig. 2: Network lifetime

Fig. 3: Energy dissipation

Finally, the Fig. 4 presents the comparison of network life time using three different scenarios: i) First Node Dies (FND), ii) Half of the Nodes Dies (HND) and iii) 90% of the Node dies i.e. Most of the Nodes Dies (MND)[15]. The result shows that our scheme can increase the lifetime of a sensor network by 64% for FND, 27% for HND and 40% for MND.

Fig. 4: Comparison of network lifetime

V. CONCLUSION

In this paper, energy efficient clique based clustering and routing (CBCR) protocol which discovers its connected neighbor sets and initiates clustering process on the basis of maximum degree of connectivity. Moreover, CBCR selects maximum energy cluster head for each round depending on the pre-defined threshold level. The proposed protocol can be widely used in data-driven application, data fusion, routing and implementing security mechanisms. Our future plan is to implement CBCR protocol in dynamic scenario to find out effects of node mobility, message collisions, network delay etc. Moreover, we would like to implement security mechanisms to prevent malicious attacks that are very common to wireless sensor networks.

REFERENCES

1. Ameer Ahmad Abbasi, M.Y., *A Survey on Clustering Algorithms for Wireless Sensor Networks*. Journal of Computer Communications, Special Issue on Network Coverage and Routing Schemes for Wireless Sensor Networks, 2007. **30**: p. 2826–2841.
2. *Habitant Monitoring on Great Duck Island*. 5 Jan, 2013; Available from: http://www.iwi.uni-hannover.de/lv/seminar_ss04/www/Jan_Gacnik/bibliography/greatduck.pdf.

3. W.B. Heinzelman, A.P.C., H. Balakrishnan, *Application specific protocol architecture for wireless microsensor networks*. IEEE Transactions on Wireless Networking, 2002.
4. A. Manjeshwar, D.A. *TEEN: a Routing Protocol for Enhanced Efficient in Wireless Sensor Networks*. in *Proc. of the 15th International Parallel and Distributed Processing Symposium*. 2001.
5. S. Lindsey, C.R. *PAGASIS: Power Efficient Gathering in Sensor Information System*. in *Proc. of 2002 IEEE Aerospace Conference*. March, 2002.
6. S. Yi, J.H., Y. Cho, J. Hong, *PEACH: Power-efficient and adaptive clustering hierarchy protocol for wireless sensor networks*. Computer Communications, Elsevier, 2007. **30**(14-15): p. 2842-2852.
7. R. Chang, C.K., *An energy efficient routing mechanism for Wireless Sensor Networks*, in *20th International Conference on Advanced Information Networking and Applications (AINA '06)*Apr. 2006.
8. O. Younis, S.F., *HEED: A Hybrid, Energy-Efficient, Distributed clustering approach for Ad Hoc sensor networks*. IEEE Transactions on Mobile Computing 2004. **3**(4): p. 366–379.
9. S. Bandyopadhyay, E.C. *An energy efficient hierarchical clustering algorithm for wireless sensor networks*. in *Proceedings of the 22nd Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM 2003)*. April 2003. San Francisco, California.
10. Agha, T.P.a.G. *Maximal clique based distribution group formation for autonomous agent coalitions*. in *3rd Int'l joint Conf. on Agents and Multi Agent Systems*. 2004.
11. K. Sun, P.P., P. Ning, *Secure Distributed Cluster Formation in Wireless Sensor Networks in 22nd Annual Computer Security Applications Conference, ACSAC '06*December 2006: Miami Beach, FL, USA. p. 131 - 140
12. Alain Bertrand Bomgni, J.F.M., *An Energy-Efficient Clique-Based Geocast Algorithm for Dense Sensor Networks*, in *Communications and Network*, 2010, SciRes. p. 125-133.
13. M. Garey, D.J., *Computers and Intractability: A Guide to the Theory of NP-Completeness*1979: W. H. Freeman and Company.
14. Dharwadker, A., *The Clique Algorithm*Oct.,2011: Amazon. 46.